ACM Conference on Computer and Communications Security, CCS, pp. 272–280, 2003.

[89] E. G. Barrantes, D. H. Ackley, T. S. Palmer, D. Stefanovic, and D. D. Zovi, "Randomized Instruction Set Emulation to Disrupt Binary Code Injection Attacks," in *Proceedings of the 10th ACM Conference on Computer and Communications Security, CCS*, pp. 281–289, 2003.

[90] M. Chew and D. Song, "Mitigating Buffer Overflows by Operating System Randomization," Tech. Rep. CS-02-197, Carnegie Mellon University, 2002.

[91] D. Couroussé, T. Barry, B. Robisson, P. Jaillon, O. Potin, and J. Lanet, "Runtime Code Polymorphism as a Protection Against Side Channel Attacks," in *Proceedings of Information Security Theory and Practice - 10th IFIP WG 11.2 International Conference, WISTP*, vol. 9895, pp. 136–152, 2016.

[92] M. Jacob, M. H. Jakubowski, P. Naldurg, C. W. Saw, and R. Venkatesan, "The Superdiversifier: Peephole Individualization for Software Protection," in *Proceedings of Advances in Information and Computer Security, Third International Workshop on Security, IWSEC 2008*, vol. 5312, pp. 100–120, 2008.

[93] M. Henry, "Superoptimizer: A Look at the Smallest Program," *ACM SIGARCH Computer Architecture News*, vol. 15, pp. 122–126, Nov 1987.

[94] V. Le, M. Afshari, and Z. Su, "Compiler Validation via Equivalence Modulo Inputs," in *ACM SIGPLAN Conference on Programming Language Design and Implementation, PLDI*, pp. 216–226, 2014.

[95] B. R. Churchill, O. Padon, R. Sharma, and A. Aiken, "Semantic Program Alignment for Equivalence Checking," in *Proceedings of the 40th ACM SIGPLAN Conference on Programming Language Design and Implementation, PLDI*, pp. 1027–1040, 2019.

[96] V. Le, M. Afshari, and Z. Su, "Compiler Validation via Equivalence Modulo Inputs," in *ACM SIGPLAN Conference on Programming Language Design and Implementation, PLDI*, pp. 216–226, 2014.

[97] N. Harrand, C. Soto-Valero, M. Monperrus, and B. Baudry, "Java Decompiler Diversity and Its Application to Meta-decompilation," *J. Syst. Softw.*, vol. 168, p. 110645, 2020.

[98] M. Zalewski, "American Fuzzy Lop," 2017.

[99] K. Zhang, D. Wang, J. Xia, W. Y. Wang, and L. Li, "ALGO: Synthesizing Algorithmic Programs with Generated Oracle Verifiers," *CoRR*, vol. abs/2305.14591, 2023.

[100] L. de Moura and N. Bjørner, "Z3: An Efficient SMT Solver," in *Tools and Algorithms for the Construction and Analysis of Systems*, (Berlin, Heidelberg), pp. 337–340, 2008.

[101] A. Abate, C. David, P. Kesseli, D. Kroening, and E. Polgreen, "Counterexample Guided Inductive Synthesis Modulo Theories," in *Proceedings of Computer Aided Verification - 30th International Conference, CAV*, vol. 10981, pp. 270–288, 2018.

[102] P. M. Phothilimthana, A. Thakur, R. Bodík, and D. Dhurjati, "Scaling up Superoptimization," in *Proceedings of the Twenty-First International Conference on Architectural Support for Programming Languages and Operating Systems, ASPLOS*, pp. 297–310, 2016.

[103] R. El-Khalil and A. D. Keromytis, "Hydan: Hiding Information in Program Binaries," in *Information and Communications Security, 6th International Conference, ICICS*, vol. 3269, pp. 187–199, 2004.

[104] V. Singhal, A. A. Pillai, C. Saumya, M. Kulkarni, and A. Machiry, "Cornucopia : A Framework for Feedback Guided Generation of Binaries," in *37th IEEE/ACM International Conference on Automated Software Engineering, ASE 2022, Rochester, MI, USA, October 10-14, 2022*, pp. 27:1–27:13, ACM, 2022.

[105] B. Cox and D. Evans, "N-Variant Systems: A Secretless Framework for Security through Diversity," in *Proceedings of the 15th USENIX*, 2006.

[106] D. Bruschi, L. Cavallaro, and A. Lanzi, "Diversified Process replicæ for Defeating Memory Error Exploits," in *Proceedings of the 26th IEEE International Performance Computing and Communications Conference, IPCCC 2007, April 11-13, 2007, New Orleans, Louisiana, USA*, pp. 434–441, IEEE Computer Society, 2007.

[107] B. Salamat, A. Gal, T. Jackson, K. Manivannan, G. Wagner, and M. Franz, "Stopping Buffer Overflow Attacks at Run-Time: Simultaneous Multi-variant Program Execution on a Multicore Processor," tech. rep., Technical Report 07-13, School of Information and Computer Sciences, UCIrvine, 2007.

[108] L. Davi, C. Liebchen, A. Sadeghi, K. Z. Snow, and F. Monrose, "Isomeron: Code Randomization Resilient to (Just-In-Time) Return-oriented Programming," in *22nd Annual Network and Distributed System Security Symposium, NDSS 2015, San Diego, California, USA, February 8-11, 2015*, The Internet Society, 2015.

[109] G. Agosta, A. Barenghi, G. Pelosi, and M. Scandale, "The MEET Approach: Securing Cryptographic Embedded Software Against Side Channel Attacks," *IEEE Trans. Comput. Aided Des. Integr. Circuits Syst.*, vol. 34, no. 8, pp. 1320–1333, 2015.